

LAPPED BIORTHOGONAL TRANSFORMS FOR TRANSFORM CODING WITH REDUCED BLOCKING AND RINGING ARTIFACTS

Henrique S. Malvar

PictureTel Corporation
100 Minuteman Road M/S 635, Andover, MA 01810, USA
malvar@pictel.com

(Note: this is a corrected version, with a corrected Fig. 4 and PSNR values)

ABSTRACT

Two new lapped transforms are introduced: the LBT (lapped biorthogonal transform) and the HLBT (hierarchical lapped biorthogonal transform). The LBT has the same computational complexity of the LOT (lapped orthogonal transform), with much less blocking artifacts. The HLBT has a significantly lower computational complexity than the LOT, essentially no blocking artifacts, and less ringing artifacts than the commonly-used DCT (discrete cosine transform). The LBT and HLBT have a transform coding gain that is typically between 0.5 and 1.2 dB higher than that of the DCT. Image coding examples using JPEG and embedded zerotree coders demonstrate the better performance of the LBT and HLBT.

1. INTRODUCTION

Transform coding (TC) is an efficient and commonly used tool for signal compression. TC is used as a basis for many video, image, and audio coding standards, such as MPEG and JPEG. In the encoding process, the signal is divided into blocks of N samples (or $N \times N$ for images). For each block, a transform operator is applied. The resulting transform coefficients are quantized (usually, via scalar quantizers) and entropy encoded. At the decoder, the reverse operations are performed: entropy decoding, dequantization, and inverse transformation. The decoded blocks are combined to form the reconstructed signal.

Two kinds of reconstruction artifacts are typical in TC, mainly at low bit rates: blocking (or tiling) and ringing. Blocking artifacts arise because the concatenation of the reconstructed blocks generates signal discontinuities across block boundaries. Ringing artifacts arise because TC reconstructs blocks by linear combinations of the transform basis functions, and the quantization errors on the transform coefficients generate signal errors that last for the entire block duration (quite similarly to the Gibbs phenomenon in truncated Fourier series).

Lapped transforms (LTs) [1, 2] reduce blocking effects significantly, because their basis functions have two key properties: (i) they are longer than the block size, and (ii) they decay smoothly to zero at their boundaries. This is particularly true with biorthogonal solutions [2], in which the synthesis basis functions are smoother than the analysis ones. Furthermore, LTs have

higher coding gains, which lead to lower reconstruction error energies. However, the longer basis functions of LTs cause an increase in ringing artifacts.

In this paper we introduce two new lapped transforms: the lapped biorthogonal transform (LBT), which is both a fast-computable approximation of the biorthogonal filter banks in [2] and a simpler version of the GLTs in [3], and the hierarchical lapped biorthogonal transform (HLBT), which uses the hierarchical constructions in [4] and [5] applied to the LBT. The HLBT leads to less blocking than lapped orthogonal transforms (LOTs) and less ringing than the discrete cosine transform (DCT), while maintaining a computational complexity lower than that of the LOT.

2. THE LBT

The direct and inverse LBT are defined by the flowgraph shown in Fig. 1, which is based on the LOT flowgraph of [1]. The only difference from the LOT flowgraph in [1] is that the first odd-symmetrical DCT coefficients (i.e., the first AC coefficients) are multiplied by $\sqrt{1/2}$ in the inverse transform (synthesis), and by $\sqrt{2}$ in the direct transform (analysis). This is represented in Fig. 1 by the matrix V , defined by $V = \text{diag}\{\sqrt{2}, 1, 1, \dots, 1\}$ for the direct transform, and by $V^{-1} = \text{diag}\{\sqrt{1/2}, 1, 1, \dots, 1\}$ for the inverse transform. The construction of the LBT in Fig. 1 is quite similar to that in [5], where the DC coefficients were scaled, and in [3], where all coefficients were scaled.

3. THE HLBT

The HLBT is a two-level hierarchical lapped transform [4], built from the LBT as shown in Figure 2. It is similar to the three-level HMFLT of [5], which is based on the MFLT (modified fast lapped transform) [5].

From a subband decomposition viewpoint, the HLBT in Fig. 2 is a nonuniform (multiresolution) filter bank with 5 bands. The first two subbands, corresponding to coefficients with indices 0 and 1, have bandwidth $\pi/8$. The third subband covers $\pi/4$ to $\pi/2$, corresponding to coefficients 2 and 3, and so forth.

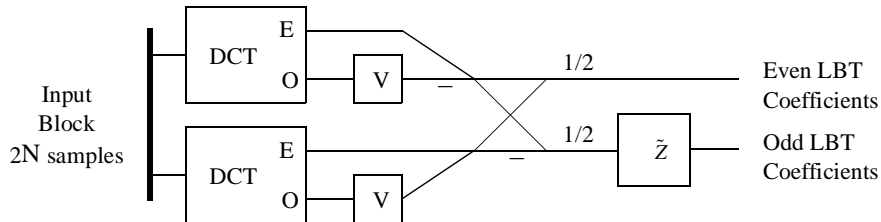


Figure 1: Flowgraph of the LBT; direct transform from left to right, inverse from right to left.

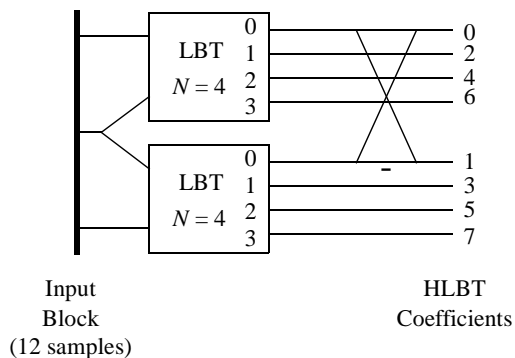


Figure 2: HLBT flowgraph for $N = 8$.

The LBT basis functions are shown in Fig. 3 (a) and (b), where it is clear the reason for the scaling matrix V in Fig. 1: increased smoothness of the synthesis functions. Unlike the LOT, where the basis functions decay to a value near zero at the boundaries [1], the LBT synthesis basis functions decay to zero at the boundaries. That is why the LBT has less blocking artifacts than the LOT, as shown in Section 5. Although the analysis functions approach relatively large nonzero values at their boundaries, that does not contribute to blocking artifacts.

The HLBT basis functions of Fig. 3(c) have many interesting properties. The low-frequency functions (DC and first AC) have an overlap of 33% (compared to 50% for the LOT and LBT). The high-frequency functions (indices 3 to 7) have the same length as the DCT basis functions (equal to the block size $N = 8$). However, since the HLBT functions decay to zero smoothly, their effective length is actually less than the block size, and that is why they generate less ringing than the DCT, LBT or LOT, as shown in Section 5.

4. CODING GAIN AND COMPUTATIONAL COMPLEXITY

The transform coding gain [1–5] is a common measure of transform performance in TC. It is usually computed for high bit rates

only, as the ratio of the arithmetic to the geometric means of the transform coefficient variances, and is usually expressed in dB. In general, the coding gain is defined as the improvement in SNR (signal-to-noise ratio) over straight PCM quantization of the original signal, and so a low bit rate TC can be computed by actually performing bit allocation [6,7], evaluating the resulting SNR, and computing its ratio to that of PCM. Low-rate TCs are more meaningful in predicting practical transform coding performance.

Considering a Gauss-Markov input signal with an inter-sample correlation coefficient of 0.95 (a common model for image coding [6,7]), the coding gains of the LOT, LBT, and HLBT are shown in Table 1, in terms of improvements over the DCT.

Also shown in Table 1 are the computational complexities of the new biorthogonal lapped transforms, obtained from the flowgraphs in Figs. 1 and 2. The DCT and LOT complexities are those reported in [7]. The LBT coding gain is within less than 0.1 dB of the optimal lapped biorthogonal transform (not fast-computable) of [2].

Table 1: Coding gain above DCT and computational complexity for $N = 8$ and a bit rate of 0.3 bits/sample.

Transform	Coding Gain (dB)	Multiplies	Adds
DCT	0	13	29
LOT	0.40	22	54
HLBT	0.65	16	42
LBT	1.05	23	54

5. IMAGE CODING EXAMPLES

We tested the transforms in Table 1 with two DCT-based image coding algorithms: the popular JPEG coder [8], and the recently introduced embedded zerotree DCT coder [9] (which replaces the wavelet transform of the SPIHT coder in [10] by DCTs, with appropriate coefficient ordering). For the JPEG coder we used a flat quantization matrix, for optimum SNR performance (the recommended quantization matrix in [8], used in almost all JPEG implementations, was designed for good visual appearance, not maximum SNR). For both coders, we tested the lapped transforms by simply replacing the DCT by the LOT, LBT, and HLBT, without any other change in the coder.

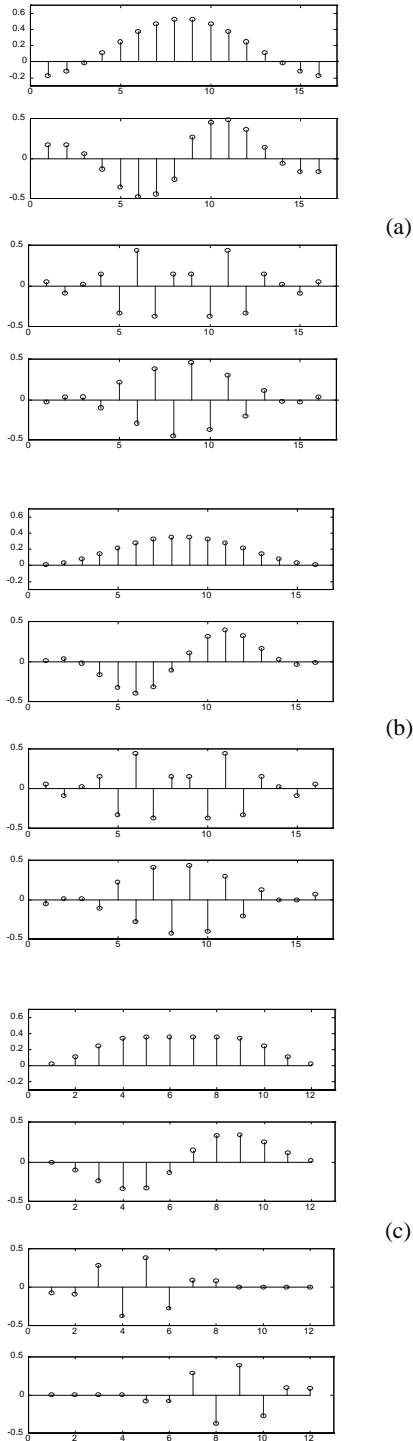


Figure 3: Basis functions of lapped biorthogonal transforms for $N = 8$, with indices 0, 1, 6, and 7: (a) LBT, analysis; (b) LBT, synthesis; and (c) HLBT, synthesis.

As a test image, we used “lena2,” a 256x256 cut from the well known 512x512 “lena” image. The PSNR (peak SNR) results are shown in Fig. 4, for rates between 0.15 and 1.0 bit per sample. For the JPEG coder, the HLBT shows a small PSNR improvement over the DCT (around 0.3 dB), whereas the LBT shows a PSNR gain of more than 0.8 dB at rates around 0.5 bits/sample. These results are consistent with Table 1.

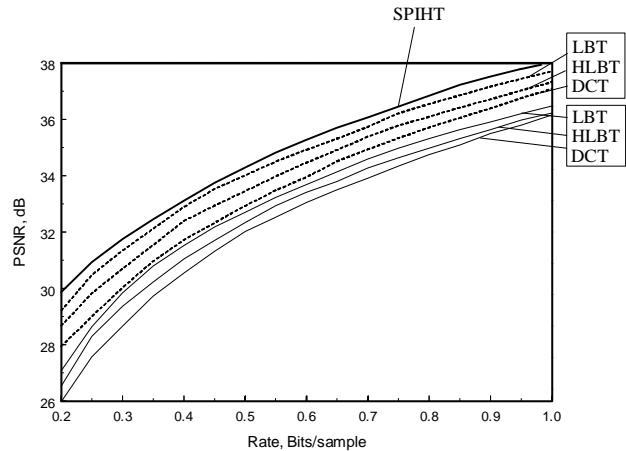


Figure 4: Performance for transform-based coders with $N = 8$, for the “lena2” image. Dashed lines: embedded zerotree; solid lines: JPEG. Reference: SPIHT (top solid line).

It is interesting to note that the performance of the HLBT is quite close to that of the LBT for the embedded zerotree coder. Both perform closely (within 0.8 and 0.4 dB, respectively) to the optimized wavelet-based SPIHT coder (one of the best image coders reported to date), but the HLBT-based embedded coder is faster. The HLBT improves de embedded coder by about 0.5 dB when compared to the DCT.

Fig. 5 shows 128x128 portions of the reconstructed images, for the rate of 0.6 bits/sample. In the top row (JPEG results), we see that all lapped transforms have less blocking than the DCT. The LOT still shows some artifacts, and the LBT is virtually free from blocking, but both show more ringing artifacts than the DCT. The HLBT has less blocking and less ringing than the DCT.

The embedded zerotree coded images in the bottom row of Fig. 5 show the significant improvement achieved with the bi-orthogonal lapped transforms. The HLBT- and LBT-coded images are quite similar, as expected from the curves in Fig. 4, and they both represent a visible improvement over the DCT-coded image.

6. CONCLUSION

We introduced two new lapped biorthogonal transforms, the LBT and the HLBT, which are good alternatives to the DCT and LOT for transform image coding. These transforms are obtained from modifications in the LOT flowgraph, similar to the ideas in [3,5]. The LBT has a performance close to the optimal bi-orthogonal lapped transform of [2].

The HLBT has a better coding gain than the DCT. Also, because of its smooth low-frequency functions and shorter high-frequency functions, it leads to less blocking and less ringing artifacts than the DCT. These advantages come at a computational overhead of only 30% (compared to 80% for the LOT or LBT).

Specifically for the embedded zerotree coder, replacing the DCT by the HLBT can improve the PSNR performance by more than 1 dB. An HLBT-based embedded zerotree coder represents a significant improvement over the DCT-based JPEG, with less blocking and ringing artifacts, an embedded bitstream, and single-pass encoding for any desired target bit rate.

REFERENCES

- [1] H. S. Malvar and D. H. Staelin, "The LOT: transform coding without blocking effects," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 37, Apr. 1989, pp. 553–559.
- [2] S. O. Aase and T. A. Ramstad, "On the optimality of non-unitary filter banks in subband coders," *IEEE Trans. on Image Processing*, vol. 4, Dec. 1995, pp. 1585–1591.
- [3] S. C. Chan, "The generalized lapped transform (GLT) for subband coding applications," *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Detroit, May 1995, pp. 1508–1511.
- [4] H. S. Malvar, "Efficient signal coding with hierarchical lapped transforms," *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Albuquerque, Apr. 1990, pp. 1519–1522.
- [5] R. W. Young and N. G. Kingsbury, "Video compression using lapped transforms for motion estimation/compensation and coding," *Proc. SPIE Conf. on Visual Communications and Image Processing*, Boston, Nov. 1992, pp. 276–288.
- [6] N. S. Jayant and P. Noll, *Digital Coding of Waveforms*. Englewood Cliffs, NJ: Prentice Hall, 1984.
- [7] H. S. Malvar, *Signal Processing with Lapped Transforms*. Norwood, MA: Artech House, 1992.
- [8] W. B. Pennebaker and J. L. Mitchell, *JPEG Still Image Data Compression Standard*. New York: Van Nostrand Reinhold, 1992.
- [9] Z. Xiong, O. G. Guleryuz, and M. T. Orchard, "A DCT-based embedded image coder," *IEEE Signal Processing Letters*, vol. 3, Nov. 1996, pp. 289–290.
- [10] A. Said and W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 6, June 1996, pp. 243–250.



Figure 5: Coding examples for image "lena2" at 0.6 bits/pixel. Top row: JPEG encoding; column order, left to right: DCT, LOT, LBT, HLBT, with PSNR (dB) = 33.0, 33.3, 33.7, and 33.4, respectively. Bottom row: embedded zerotree encoding; column order, left to right: DCT, LOT, LBT, HLBT, with PSNR (dB) = 34.0, 34.5, 34.9, and 34.5.